CHARITY DONORS PREDICTION

A Project Report

Submitted in the partial fulfillment of the

requirements for theaward of the degree of

Bachelor of Technology in

Department of Computer Science and

Engineering

By
180030385 T.RAMGOPAL
Under the supervision of
Mrs.V. Bhavani
Ass. Professor

Department of Computer Science and Engineering

K L E F, Green Fields,

Vaddeswaram- 522502, Guntur(Dist),
Andhra Pradesh, India.

2020-2021

2020-2021

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Declaration

The Project Report entitled **"CHARITY DONORS PREDICTION"** is a record of bonafide work of T.Ramgopal(180030385), submitted in partial fulfillment for the award of **Bachelor of Technology** in **Computer Science and Engineering** to the **Koneru Lakshmaiah Education Foundation** during the academic year 2021-2022. The results embodied in this report have not been copied from any other departments/University/Institute.

T Ramgopal 180030385

KONERU LAKSHMAIAH EDUCATION FOUNDATION

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Certificate

This is to certify that the Project Report entitled **"CHARITY DONORS PREDICTION"** is being submitted by T.Ramgopal(180030385) submitted in partial fulfillment for the award of **Bachelor of Technology** in **Computer Science and Engineering** to the **Koneru Lakshmaiah Education Foundation** is a record of Bonafide work carried out under our guidance and supervision during the academic year 2021-2022.

The results embodied in this report have not been copied from any other departments/University/Institute.

Signature of the Supervisor
Mrs.V Bhavani
Ass.Professor

Signature of the HOD
Mr. Hari Kiran Vege
Professor

Signature of the Examiner

Project Team Number:170

SNO	Student Reg No	Student Name
	180030385	T.Ramgopal

Guide Empid	Guide Name
5250	V.Bhavani

Project Title: **Evaluation Date:**

Criteria	Evidence of fulfillment (Filled by Project Team)	Remarks	Evaluation (Total 40 mark)
Engineering Design* (5) **(Contribution)** • Member 1 • Member 2 • Member 3			
Realistic Constraints+(5) • Time (Execution) • Space (Memory) • Platform • Boundary Conditions			
Standards (IEEE, ANSI, etc) (5) • Hardware • Software			
Maintainability (5) • Scalability • Environment			
Ethical, Social and Professional Issues (5) • Security • Privacy • New Method/Language • Social Applicability			
Documentation (5)			
Presentation (10)			

Internal Examiner
Signature with Full Name

Project Team Number:170

SNO	Student Reg No	Student Name
	180030385	T.Ramgopal

Guide Empid	Guide Name
5250	V.Bhavani

Project Title:

Evaluation Date:

Criteria	Evidence of fulfillment (Filled by Project Team)	Remarks	Evaluation (Total 40 mark)
Engineering Design* (5) (Contribution) • Member 1 • Member 2 • Member 3			
Realistic Constraints+(5) • Time (Execution) • Space (Memory) • Platform • Boundary Conditions			
Standards (IEEE, ANSI, etc) (5) • Hardware • Software			
Maintainability (5) • Scalability • Environment			
Ethical, Social and Professional Issues (5) • Security • Privacy • New Method/Language • Social Applicability			
Documentation (5)			
Presentation (10)			

External Examiner
Signature with Full Name

Acknowledgement

Apart from the effects of me, the success of any work depends largely on the encouragement and guidelines of many others. I take opportunity to express my gratitude to the people who have been instrumental in the successful completion of this thesis.

We would like to thank Ms. **V. Bhavani, Professor, Department of Computer Science and Engineering** for guiding and helping us in our work. We would also like to thank everyone who helped us and supported us.

We express our gratitude to **Dr. Hari Kiran Vege sir Head of the Department for Computer Science and Engineering** for providing us with adequate facilities, ways and means by which we are able to complete this project.

Last but not the least, we thank all Teaching and Non-Teaching Staff of our department and especially my classmates and my friends for their support in the completion of our project.

T.Ramgopal 180030385

INDEX

ABSTRACT

Presently a day the development of information is more fast and information is produced in an enormous scope consistently, the utilization of information mining has additionally expanded. Information Mining is the method involved with extricating and finding designs in the enormous informational collections and utilize characterization strategies for the future expectations. There is one charity called charity which feeds homeless people, for donations this charity sends postal mails to residents of Delhi requesting to donate for a cause, from the historical data it is evident that residents who earn more than 50 thousand dollars per annum are more likely to donate. But the charity cannot be able to figure out how to send postal mails to those who most likely to donate to charity and avoid sending postal mails to those who are most likely are not going to donate to charity so that charity can save much money. We have partitioned our activities into 2 modules and in every modules, we will perform activities like pre-handling and in different modules undertakings like preparation and executing the grouping models that we want to use in our ventures and in the following stage we will utilize perception and show how exact the models produce results. At First we will perform ETL activities and concentrate information utilizing the data set the data set server has a place with amazon Aws and afterward we will perform pre-handling on the information and eliminate any loud information and afterward we will isolate the informational collection into preparing informational index and testing informational collection and apply grouping and measure the exactness and review score and afterward we will utilize perception and plot the roc bend with the assistance of Roc bend we will actually want to decide the presentation of the Classification models we constructed. We assess the proposed technique on a few picture datasets and face datasets, and the trial results show that our proposed strategy performs better compared to other cutting edge learning calculations.

Keywords: Data Mining, ETL, Classification, Data Visualization.

Introduction:

For the most part, information mining (once in a while called information or information disclosure) is the method involved with investigating information according to alternate points of view and summing up it into accommodating information - information that can be used to fabricate pay, decreases costs, or both.. Information mining programming is one of various scientific apparatuses for investigating information. It permits clients to investigate information from various aspects or points, arrange it, and sum up the connections recognized. Actually, information mining is the most common way of tracking down connections or examples among many fields in huge social data sets. An information stockroom is a social data set that is intended for question and investigation rather than for exchange handling. It typically contains authentic information got from exchange information, yet it can incorporate information from different sources. It isolates examination responsibility from exchange responsibility and empowers an association to solidify information from a few sources. Notwithstanding a social data set, an information stockroom climate incorporates an extraction, transportation, change, and stacking (ETL) arrangement, an internet based logical handling (OLAP) motor, customer investigation apparatuses, and different applications that deal with the most common way of get-together information and conveying it to business clients.

Problem Statement:

There is one charity called charity which feeds homeless people, for donations this charity sends postal mails to residents of Delhi requesting to donate for a cause, from the historical data it is evident that residents who earn more than 50 thousand dollars per annum are more likely to donate. But the charity cannot be able to figure out how to send postal mails to those who most likely to donate to charity and avoid sending postal mails to those who are most likely are not going to donate to charity so that charity can save much money.

Information mining has turned into a famous innovation in flow research and for clinical space applications. Information mining strategy includes the utilization of complex information examination instruments to find already obscure, substantial examples and connections in enormous informational index. These devices can incorporate measurable models, numerical calculation and AI strategies in early location of malignant growth. In characterization learning, the learning plan is given a bunch of ordered models from which it is relied upon to get familiar with a method of grouping inconspicuous models.

LITERATURE SURVEY

Publisher	Dataset	Algorithm	Accuracy	Classified as
Nagaraju Kolla, M.	algorithm are executed for the 265 sample dataset.	Decision tree	About 90%	Value or no value
		K-NN	About 91%	
		Logistic regression	About 94%	
Tao Jiang	algorithm are executed for the 1845sample dataset.	Naïve Bayes	85 %	Types of people Describes as active moderate and inactive in sports
		SVM	78%	
		Random Forest Tree	91%	
Breiman, L	572 olive oils were analyzed for their content of eight fatty acid	Regression Tree	89%	Types of oils which are healthy and unhealthy
		Classification		
		Logistic Regression	84%	
Kivinen, J., Mannila	Data set from National pollution authority	Classification and Regression Tree[CART]	83.87%	Vehicle is polluting or not
		Random Forest Tree[RFT]	93.54%	

. Mingers, J	Food security form government	Regression tree	85%	Types of Food which describes as weather food is healthy or not
		Random forest	85%	
		Navie bayes	95%	

Software Requirements:

- Jupyter Notebook
- Pycharm
- Mysql

The key python libraries required are as follows:

- Pandas 0.23.4
- Keras 2.2.4
- Numpy 1.15.2
- Seaborn 0.9.0
- Scikit_learn 0.20.2
- TensorFlow 2.5.0

Hardware Requirements:

Processor	:	Intel R Core I5 — 2410M
2.30GHz Disk Space	:	1 T.B
Memory	:	8 GB RAM
Operating System	:	Windows 8 or 10 (32 or 64 bit)

THEORETICAL ANALYSIS

Business Understanding

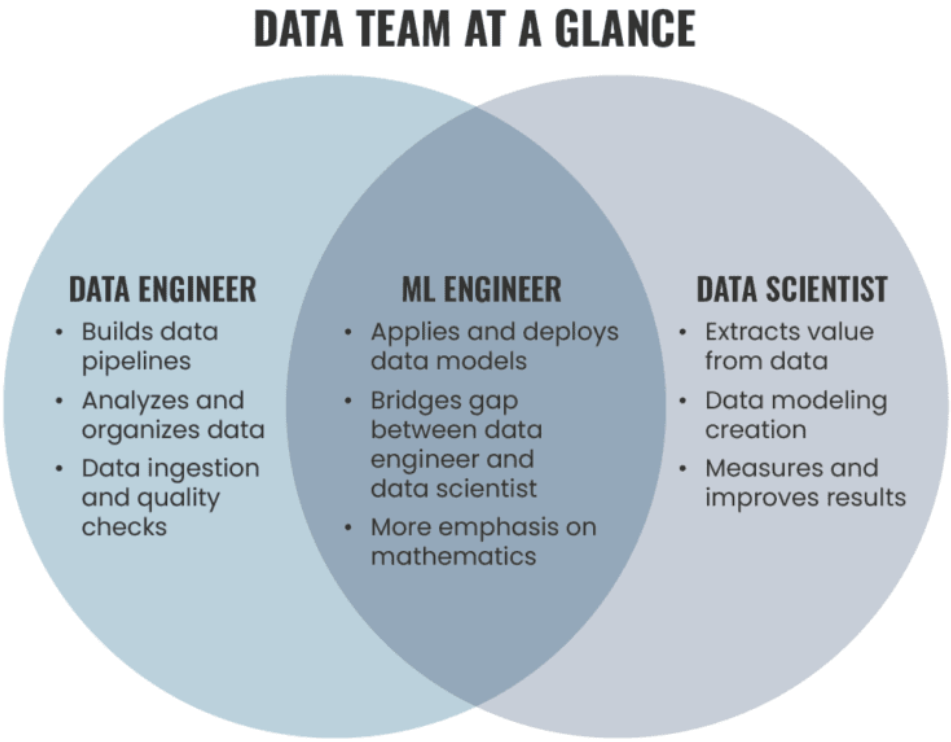
Yemego NGO is an imaginary foundation association that gives projects and administrations to veterans with spinal rope wounds or sicknesses. Direct mailing efforts are utilized to raise assets for a noble cause associations. Utilizing past mailing efforts, good cause associations can contact individuals who gave previously.

The test here is to draw experiences from past contributor history and make forecasts. By anticipating benefactors who will give and gauge how much contributors will give, we can assist with saving the assets expected to accomplish the genuine work of really focusing on the less special. One method for lessening costs for Yemego is to expand the proficiency of contributor outreach by recognizing givers probably going to give to the association. Since there is a $5 cost of mailing each mission contributor, this task plans to set aside time and cash for Yemego NGO by focusing on just the most likely potential benefactors in light of past reaction information.

Virtual communication among students and faculty is more important for better learning. The way faculty interact with students will affect their participation and training. Improvement in technology amplify the need to gain more ideas around how to deliver course materials that can improve and support the learning process. This study investigates student access patterns to educational resources available in no simultaneous online digital learning. In this project, we focus mainly on private chat with students and faculties based on their requirement. For this purpose, we are implementing a web application of online interaction system for educational institutions. For this application we use EC2 service from AWS to create an instance where we launch virtual private machine to display our project code.

Data Engineering :

Data engineers often work as part of an analytics team alongside data scientists. The engineers provide data in usable formats to the data scientists who run queries and algorithms against the information for predictive analytics, machine_learning and data mining applications. Data engineers also deliver aggregated data to business executives and analysts and other end users so they can analyze it and apply the results to improving business operations.



Data Extraction

Data extraction refers to the process of procuring data from a given source and moving it to a new context, either on-site, cloud-based, or a hybrid of both. There are various strategies employed to this end, which can be complex and are often performed manually. Unless data is extracted solely for archival purposes, it is generally the first step in the ETL process of Extraction, Transformation, and Loading. This means that after initial retrieval, data nearly always undergoes further processing in order to render it usable for future analysis.

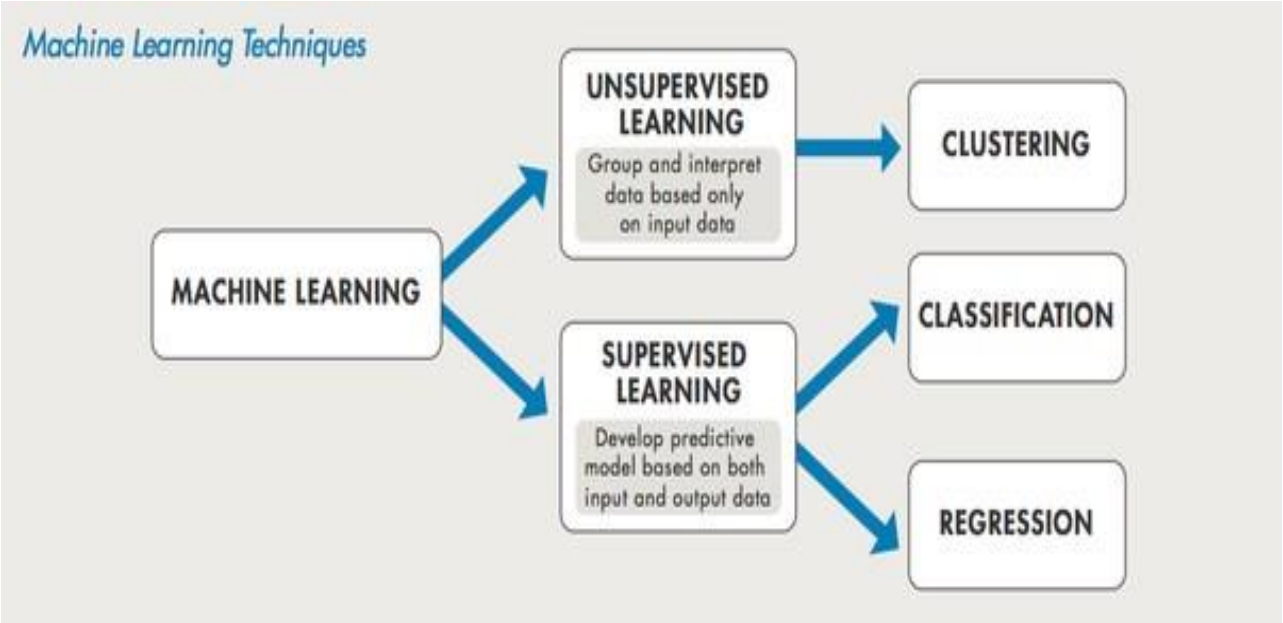
Despite the availability of highly valuable data, one survey found that organizations ignore up to 43% of accessible data. Worse yet, of the data they do collect, a mere 57% is utilized. Why is this a reason for concern?

Without a way to extract all varying data types, including the poorly structured and disorganized, businesses can't leverage the full potential of information and make the right decisions.

Working with a good dataset is crucial to ensure that your machine learning model performs well, so adopting a good data extraction method could bring countless benefits for your processes.

Machine Learning Overview:

Machine learning is an information investigating system that trains PCs to do what falls into place without a hitch for people and creatures which gain as a matter of fact. machine-learning calculations uses computational techniques to learn data from the information without falling on a pre-determined condition as model. The calculations regurlarly improve their exhibitions as the number of tests accessible for learning increments.



Supervised machine learning

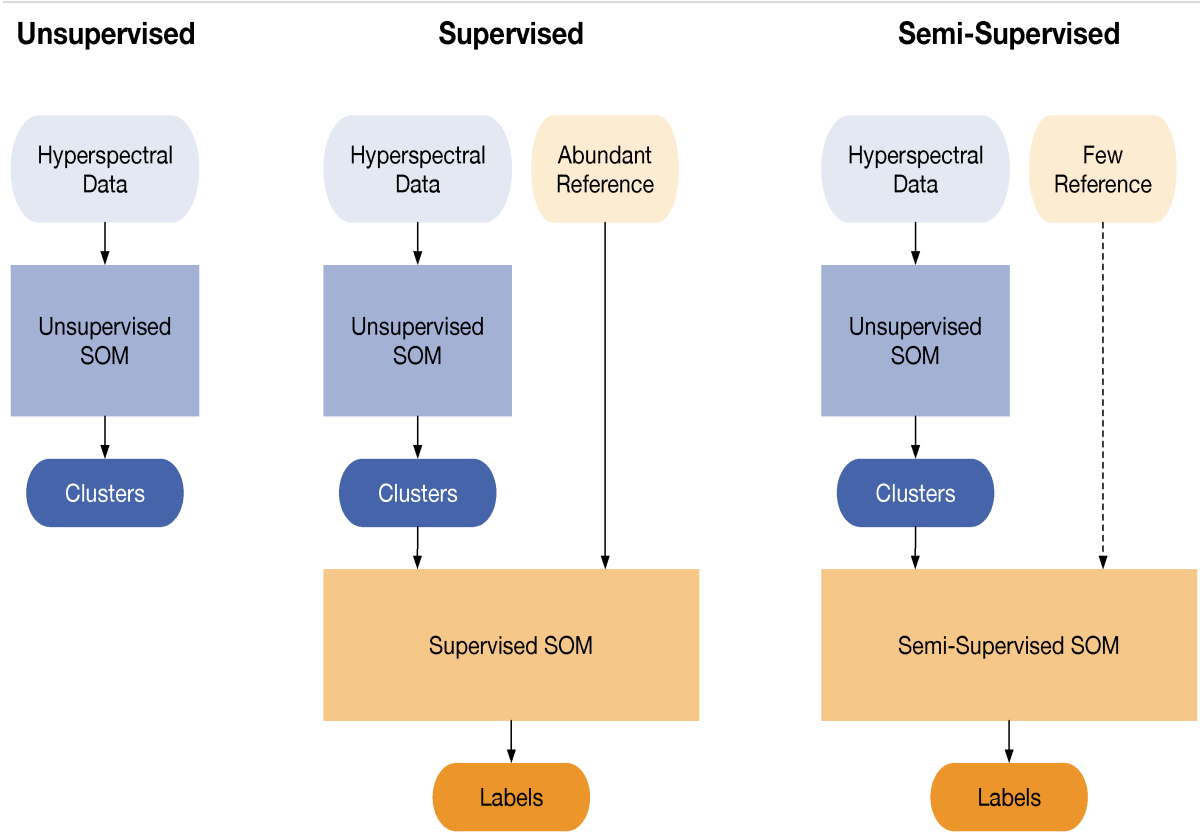
Supervised learning, also known as supervised machine learning, is defined by its use of labeled datasets to train algorithms that to classify data or predict outcomes accurately. As input data is fed into the model, it adjusts its weights until the model has been fitted appropriately. This occurs as part of the cross validation process to ensure that the model avoids overfitting or underfitting. Supervised learning helps organizations solve for a variety of real-world problems at scale, such as classifying spam in a separate folder from your inbox. Some methods used in supervised learning include neural networks, naïve bayes, linear regression, logistic regression, random forest, support vector machine (SVM), and more.

Unsupervised machine learning

Unsupervised learning, also known as unsupervised machine learning, uses machine learning algorithms to analyze and cluster unlabeled datasets. These algorithms discover hidden patterns or data groupings without the need for human intervention. Its ability to discover similarities and differences in information make it the ideal solution for exploratory data analysis, cross-selling strategies, customer segmentation, image and pattern recognition. It's also used to reduce the number of features in a model through the process of dimensionality reduction; principal component analysis (PCA) and singular value decomposition (SVD) are two common approaches for this. Other algorithms used in unsupervised learning include neural networks, k-means clustering, probabilistic clustering methods, and more.

Semi-supervised learning

Semi-supervised learning offers a happy medium between supervised and unsupervised learning. During training, it uses a smaller labeled data set to guide classification and feature extraction from a larger, unlabeled data set. Semi-supervised learning can solve the problem of having not enough labeled data (or not being able to afford to label enough data) to train a supervised learning algorithm.


Unsupervised
Supervised
Semi-Supervised
Hyperspectral Data
Unsupervised SOM
Clusters
Hyperspectral Data
Abundant Reference
Unsupervised SOM
Clusters
Supervised SOM
Labels
Hyperspectral Data
Few Reference
Unsupervised SOM
Clusters
Semi-Supervised SOM
Labels

Application of Machine Learning :

1. Virtual Personal Assistants

Machine learning is an important part of these personal assistants as they collect and refine the information on the basis of your previous involvement with them. Later, this set of data is utilized to render results that are tailored to your preferences.

Virtual Assistants are integrated to a variety of platforms. For example:

- Smart Speakers: Amazon Echo and Google Home
- Smartphones: Samsung Bixby on Samsung S8
- Mobile Apps: Google Allo

2.Predictions while Commuting

Traffic Predictions: We all have been using GPS navigation services. While we do that, our current locations and velocities are being saved at a central server for managing traffic. This data is then used to build a map of current traffic. While this helps in preventing the traffic and does congestion analysis, the underlying problem is that there are less number of cars that are equipped with GPS. Machine learning in such scenarios helps to estimate the regions where congestion can be found on the basis of daily experiences.

.

3. Videos Surveillance

Imagine a single person monitoring multiple video cameras! Certainly, a difficult job to do and boring as well. This is why the idea of training computers to do this job makes sense.

The video surveillance system nowadays are powered by AI that makes it possible to detect crime before they happen. They track unusual behaviour of people like standing motionless for a long time, stumbling, or napping on benches etc. The system can thus give an alert to human attendants,

which can ultimately help to avoid mishaps. And when such activities are reported and counted to be true, they help to improve the surveillance services. This happens with machine learning doing its job at the backend.

4. Social Media Services

From personalizing your news feed to better ads targeting, social media platforms are utilizing machine learning for their own and user benefits. Here are a few examples that you must be noticing, using, and loving in your social media accounts, without realizing that these wonderful features are nothing but the applications of ML.

- *People You May Know*
- *Face Recognition*
- *Similar Pins*

5. Email Spam and Malware Filtering

There are a number of spam filtering approaches that email clients use. To ascertain that these spam filters are continuously updated, they are powered by machine learning. When rule-based spam filtering is done, it fails to track the latest tricks adopted by spammers. Multi Layer Perceptron, C 4.5 Decision Tree Induction are some of the spam filtering techniques that are powered by ML.

.6. Product Recommendations

You shopped for a product online few days back and then you keep receiving emails for shopping suggestions. If not this, then you might have noticed that the shopping website or the app recommends you some items that somehow matches with your taste. Certainly, this refines the shopping experience but did you know that it's machine learning doing the magic for you? On the basis of your behaviour with the website/app, past purchases, items liked or added to cart, brand preferences etc., the product recommendations are made.

EXPERIMENTALANALYSIS

Data Exploration and Preprocessing The dataset I used contains a total of 14 features and information about features are as follows

age: in years(continuous)

workclass: Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked(Categorical)

education: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.(categorical)

marital-status: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouseabsent, Married-AF-spouse(categorical)

occupation: Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-houseserv, Protective-serv, Armed-Forces(categorical)

relationship: Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried(categorical)

race: Black, White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other(categorical)

sex: Female, Male.(categorical)

capital-gain: continuous.

capital-loss: continuous.

hours-per-week: continuous.

native-country: United-States, Cambodia, England, Puerto-Rico, Canada, Germany, OutlyingUS(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, ElSalvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands.

The above-described dataset contains categorical and continuous features with 45222 instances without missing values.

Descriptive features:

'age', 'workclass', 'education_level', 'education-num','marital-status', 'occupation', 'relationship', 'race', 'sex', 'capital-gain', 'capital-loss', 'hours-per-week', 'native-country',

Data information

```
In [5]: Features = df.columns
        Features
        df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45222 entries, 0 to 45221
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   age              45222 non-null  int64
 1   workclass        45222 non-null  object
 2   education_level  45222 non-null  object
 3   education-num    45222 non-null  int64
 4   marital-status   45222 non-null  object
 5   occupation       45222 non-null  object
 6   relationship     45222 non-null  object
 7   race             45222 non-null  object
 8   sex              45222 non-null  object
 9   capital-gain     45222 non-null  int64
 10  capital-loss     45222 non-null  int64
 11  hours-per-week   45222 non-null  int64
 12  native-country   45222 non-null  object
 13  income           45222 non-null  object
dtypes: int64(5), object(9)
```

DATA*:*

Firstly, we should consider the proper data with proper attributes for using it or to fit for the prediction purpose. The useful attributes by which we can predict we need toconsider all of that attributes and we need the correct outcome for that attributes we consider.

DATA PRE-PROCESSING:

Data preprocessing is a significant advance in the information mining process. The expression "trash in, trash out" is especially pertinent to information mining and AI ventures. Data gathering techniques are frequently inexactly controlled, coming about in out-of-extend values (e.g., Income: −100), outlandish information blends (e.g., Sex: Male, Pregnant: Yes), missing qualities, and so on. By acting as a virtual firewall, a security group controls incoming and outgoing traffic for your EC2 instances. Inbound rules regulate traffic entering your instance, whereas outbound rules regulate traffic exiting it. When launching an instance, you can define one or more security groups. Amazon EC2 uses the default security group if no security group is specified.Information readiness and sifting steps can take impressive measure of handling time. Information preprocessing incorporates cleaning, Instance determination, standardization, change, highlight extraction and choice, and so on. The result of information preprocessing is the last preparing set.

NOISY DATA:

Containing errors or outliers Noisy data (incorrect values) may come from Faulty data collection instruments, Human or computer error at data entry, Errors in data transmission.
e.g., Salary= ‒-10‖

INCONSISTENT DATA:

Containing discrepancies in codes or names, Inconsistent data may come from, Differentdata sources, Functional dependency violation (e.g., modify some linked data)
Duplicate records also need data cleaning e.g., Age=‒42‖ Birthday=‒03/07/1997‖
e.g., Was rating ‒1,2,3‖, now rating ‒A, B, C‖ e.g., discrepancy between duplicate records.

DATA PRE-PROCESSING IMPORTANCE:

Quality decisions must be based on quality data, Data warehouse needs consistent integration of quality data, Data extraction, cleaning, and transformation comprises the majority of the work of building a data warehouse
e.g., duplicate or missing data may cause incorrect or even misleading statistics.

MAJOR TASKS IN DATA PRE-PROCESSING:

DATA CLEANING:

Data cleaning is the way toward distinguishing and revising (or evacuating) degenerate or off base records from a record set, table, or database and alludes to recognizing inadequate, erroneous, off base or superfluous pieces of the information and afterward supplanting, changing, or erasing the messy or coarse information. Information purging might be performed intelligently with information wrangling apparatuses, or as bunch preparing through scripting. In the wake of purifying, an informational index ought to be predictable with other comparative informational indexes in the framework. The irregularities recognizedor evacuated may have been initially brought about by client section blunders, by debasement in transmission or capacity, or by various information word reference meanings of comparable substances in various stores. Information cleaning contrasts from information approval in that approval perpetually implies information is dismissed from the framework at section and is performed

DATA INTEGRATION:

Data integration includes joining information dwelling in various sources and giving clients a bound together perspective on them. This procedure gets huge in an assortment of circumstances, which incorporate both business, (for example, when two comparable organizations need to blend their databases) and logical (consolidating research results from various bioinformatics stores, for instance) spaces. Information joining shows up with expanding recurrence as the volume (that is, enormous information) and the need to share existing information detonates. It has gotten the focal point of broad hypothetical work, and various open issues stay unsolved.

By acting as a virtual firewall, a security group controls incoming and outgoing traffic for your EC2 instances. Inbound rules regulate traffic entering your instance, whereas outbound rules regulate traffic exiting it. When launching an instance, you can define one or more security groups. Amazon EC2 uses the default security group if no security group is specified.Every datum source is divergent and accordingly isn't intended to help dependable joins between information sources. Accordingly, information virtualization just as informationalliance relies on incidental information shared characteristic to help joining information and data from divergent informational indexes. In view of this absence of information esteem shared characteristic crosswise over information sources, the arrival set might be mistaken, fragmented, and difficult to approve.

One arrangement is to recast divergent databases to incorporate these databases without the requirement for ETL. The recast databases bolster shared characteristic requirements where referential trustworthiness might be upheld between databases. The recast databases furnish planned information get to ways with information esteem shared trait crosswise over databases.

MISSING DATA:

Data is not always available

E.g., many tuples have no recorded value for several attributes, such as customer income in sales data, missing data may need to be inferred

Missing data may be due to:

- equipment malfunction
- inconsistent with other recorded data and thus deleted
- data not entered due to misunderstanding
- certain data may not be considered important at the time of entry
- not register history or changes of the data

HANDLING MISSING DATA:

Ignore the tuple: usually done when class label is missing (assuming the tasks in classification not effective when the percentage of missing values per attribute varies considerably.

- Fill in the missing value manually: tedious + infeasible?
- Fill in it automatically with

A global constant: e.g., –unknown‖, a new class?

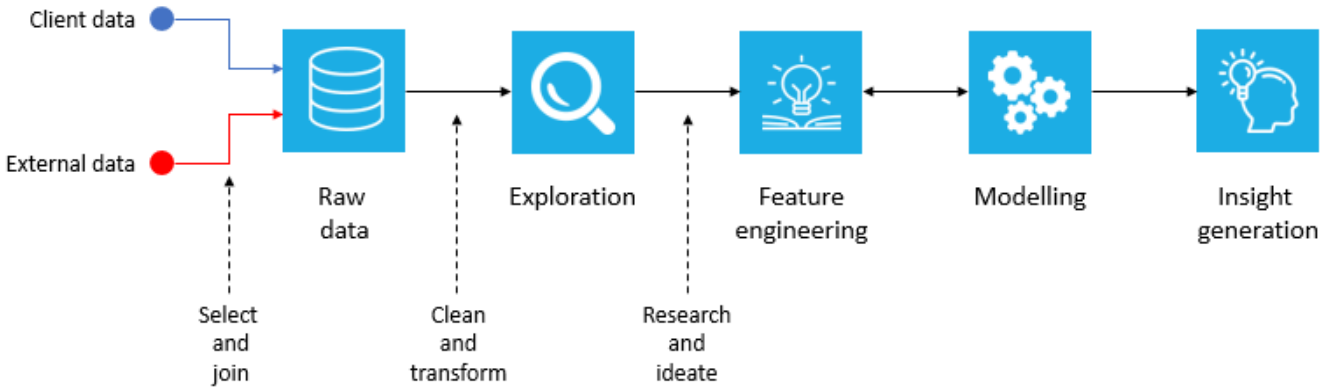
The attribute mean

The attribute mean for all samples belonging to the same class: smarter

The most probable value: inference-based such as Bayesian formula or decision tree.

I. DESIGN

Work Flow



1) *Proposed Approach Steps:*

1. First, we Extract donar dataset from Database .
2. Filter dataset according to requirements and create a new dataset which has attribute according to analysis to be done.
3. Perform Pre-Processing onthe dataset.
4. Split the data into training and testing.
5. Train the model with training data then analyze testing dataset over classification algorithms.
6. Finally you will get results as accuracy metrics.
7.

Modules

- Data collection
- Data pre-processing
- Feature extration
- Evaluation model

ALGORITHMS USED IN THE PROJECT:

I trained four models using preprocessed data, and the models are as follows

DecisionTreeClassifier(Information based learning)

KNeighborsClassifier(Similarity-Based Learning)

Naive Bayes(Probability-Based Learning)

Logistic regression(Error Based Learning)

DecisionTreeClassifier :- The inputs for model are13 descriptive features and one target feature for training . after training model is provided with sampled test data without target feature to evaluate model performance I have trained DecisionTreeClassifier with different Tree Depths and also with different impurity measures and noted and compared accuracies for Tree Depths and different impurity measures

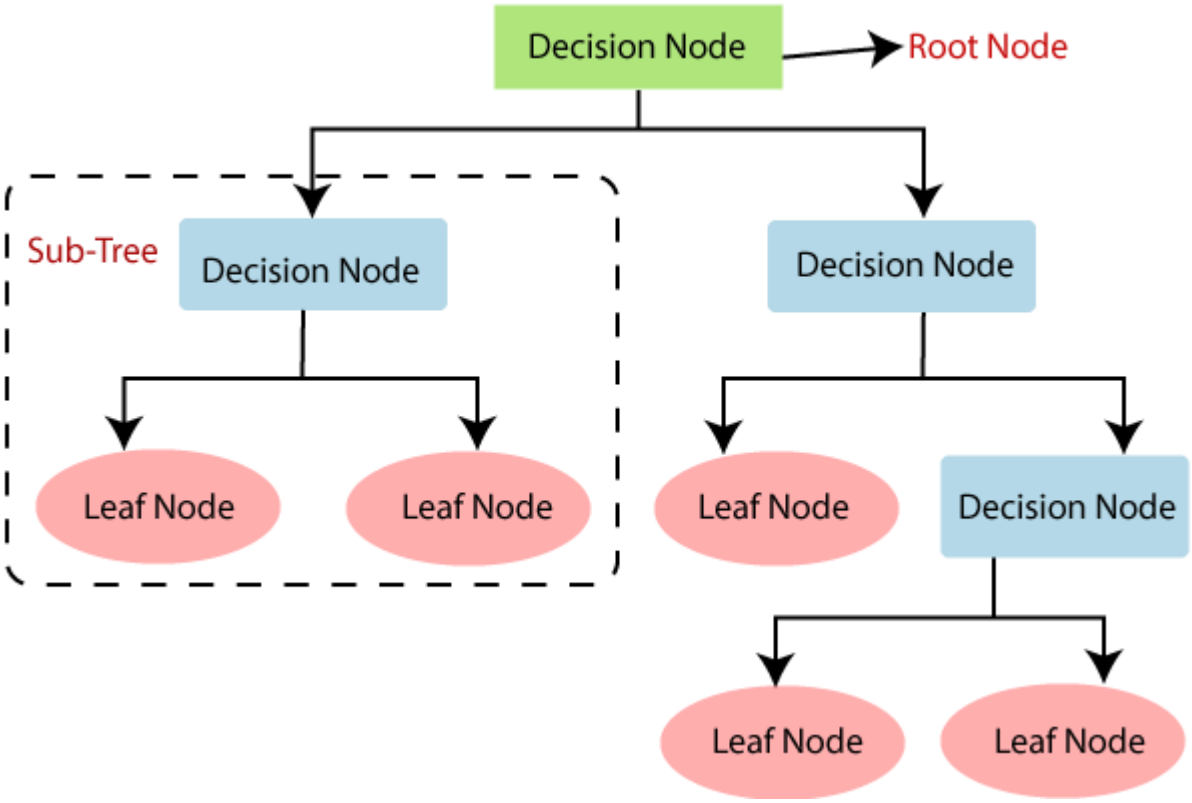
KNeighborsClassifier :- The inputs for model are13 descriptive features and one target feature for training . after training model is provided with sampled test data without target feature to evaluate model performance The model was evaluated with different k values and noted their accuracies

Naive Bayes:- The inputs for model are13 descriptive features and one target feature for training . after training model is provided with sampled test data without target feature to evaluate model performance and calculated accuracy.

Logistic regression:- The inputs for model are13 descriptive features and one target feature for training . after training model is provided with sampled test data without target feature to evaluate model performance and calculated accuracy.

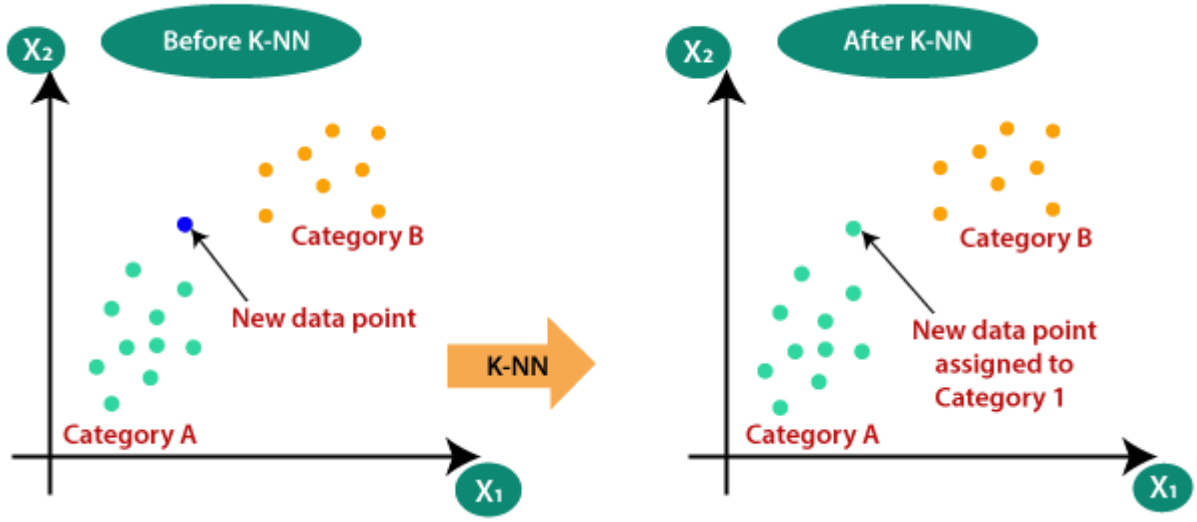
DecisionTreeClassifier:

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome. In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.The decisions or the test are performed on the basis of features of the given dataset.*It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.*It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.



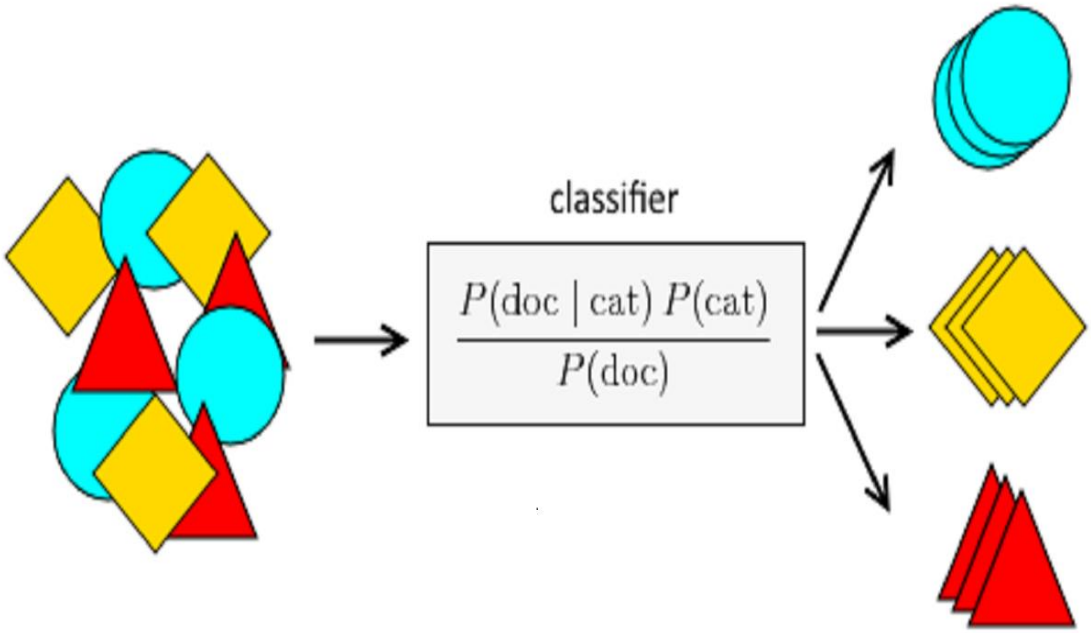
KneighborsClassifer:

KNeighborsClassifier implements classification based on voting by nearest k-neighbors of target point, t, while *RadiusNeighborsClassifier* implements classification based on all neighborhood points within a fixed radius, r, of target point, t. In *NearestCentroid* classifier, each class is represented by the centroid of its members; thus the target point will be member of that class whose centroid is nearest to it. *NearestCentroid* algorithm is the simplest of the three and has no parameters to select from. Its results can be taken as the benchmark for evaluation purposes.



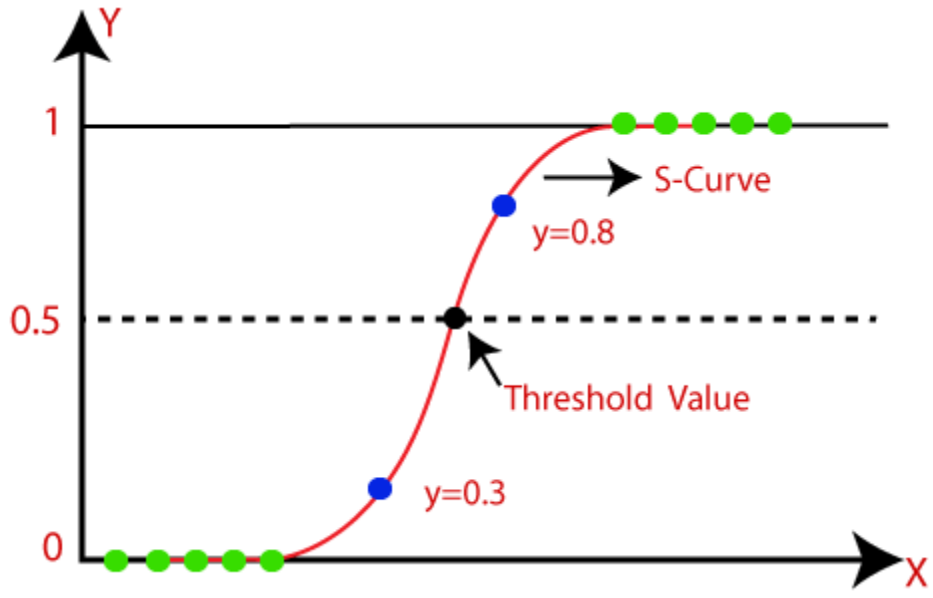
Naive Bayes:

Naive Bayes falls under the umbrella of supervised machine learning algorithms that are primarily used for classification. In this context, "supervised" tells us that the algorithm is trained with both input features and categorical outputs (i.e., the data includes the correct desired output for each point, which the algorithm should predict).



LOGISTIC REGRESSION

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems.In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.



Sample Coding

Installing Libraries :

```
from scipy import stats
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

Connecting to Database :

```
#Connecting to mysql database, AWS Server
import pymysql
conn=pymysql.connect(host='database-1.cjayyplhux0p.us-east-
2.rds.amazonaws.com',port=int(3306),user='admin',passwd='Chinnu.969',db='project')
cursor=conn.cursor()
```

Checking the Connection and Data :

```
data=cursor.execute("select * from finaltable")
for row in cursor.fetchall():
print (row)
```

Storing Data into Dataframe from Database :

```
from sqlalchemy import create_engine
import pymysql

db_connection_str = 'mysql+pymysql://admin:Chinnu.969@database-1.cjayyplhux0p.us-east-
2.rds.amazonaws.com:3306/project'
db_connection = create_engine(db_connection_str)
df = pd.read_sql('SELECT * FROM finaltable', con=db_connection)
```

Finding unique values from each column:

```
unique_values = []
features = data.columns
for feature in features :
unique = data[feature].unique().shape[0]
```

Division of categorical and continuous features:

```
Numerical_Features = data._get_numeric_data().columns
list1 = []
list1.append(Numerical_Features[1])
Continuous_Features = set(Numerical_Features) - set(list1)
Categorical_Features = set(features) - set(Numerical_Features)
Categorical_Features.update(list1)
print("Continuous features are {}".format(Continuous_Features))
```

preparation of data quality report:

```
# formation of data quality report and display the data quality report
Continuous_Features_ABT = pd.DataFrame(data = Continuous_Features,columns=['feature'] )
Continuous_Features_ABT['count'] = count_values
Continuous_Features_ABT['missing values(%)'] = missing_values_percent_list
Continuous_Features_ABT['Unique values'] =  Con_unique_values
Continuous_Features_ABT['minimum value'] = minimum_values.values
Continuous_Features_ABT['1ST Quartile'] = q1.values
Continuous_Features_ABT['mean'] = mean_values.values
Continuous_Features_ABT['median'] = median_values.values
Continuous_Features_ABT['3RD Quartile'] = q3.values
Continuous_Features_ABT['maximum value'] = maximum_values.values
Continuous_Features_ABT['standard deviation'] = Standatrd_deviation.values
print("Data Quality Report for Continuous features")
display(Continuous_Features_ABT)
```

visualization

```
for feature in features :
   print(feature)
   plt.hist(data[feature],bins = 20,align='right')
   plt.ylim((0, 10000))
   # plt.yticks([0, 500, 1000, 1500, 2000])
   plt.xticks( rotation='vertical')
   plt.show()
```

feature transformation

```
Feature_Transformed_DF = pd.get_dummies(feature_selected_DF)
Feature_Transformed_DF.info()
```

Loading preprocessed data into dataframe using panda library

```
df = pd.read_csv('R:\Ram\Computer Science\FINAL PROJECT\Main\Dataset_After__pp.csv')
df = df.replace({'income': '>50K'},1)
df = df.replace({'income': '<=50K'},0)
data = df.copy()
data = data.drop('Unnamed: 0',axis = 1)
x = data.drop('income',axis=1)
y = data['income']
df
```

splitting data into testing and training data with 40% holdout

```
X_train,X_test,y_train,y_test = train_test_split(x,y,test_size = 0.4)
```

Training and testing model with DecisionTreeClassifier with differnt TreeDepths and impurity as entropy

```
Training_score = []
Testing_score = []
Treedepth = []
rt = pd.DataFrame(columns= ['TreeDepth','Training score','Testing score'])
for TreeDepth in range(1,20) :
    clf = DecisionTreeClassifier(criterion='entropy', max_depth=TreeDepth, random_state=0)
    clf.fit(X_train,y_train)
    Training_score.append(clf.score(X_train,y_train))
    Testing_score.append(clf.score(X_test,y_test))
    Treedepth.append(TreeDepth)
rt['TreeDepth'] = Treedepth
rt['Training score'] = Training_score
rt['Testing score'] = Testing_score
rt
```

Training and testing model with DecisionTreeClassifier with differnt TreeDepths and impurity as gini

```
Training_scoreg = []
Testing_scoreg = []
Treedepthg = []
rtg = pd.DataFrame(columns= ['TreeDepth','Training score','Testing score'])
for TreeDepth in range(1,20) :
    clfg = DecisionTreeClassifier(criterion='gini', max_depth=TreeDepth, random_state=50)
    clfg.fit(X_train,y_train)
    Training_scoreg.append(clfg.score(X_train,y_train))
    Testing_scoreg.append(clfg.score(X_test,y_test))
    Treedepthg.append(TreeDepth)
rtg['TreeDepth'] = Treedepthg
rtg['Training score'] = Training_scoreg
rtg['Testing score'] = Testing_scoreg
rtg
```

Training and testing model with naivebayes GaussianNB and scores respectively stored

```
clfnv = GaussianNB()
clfnv.fit(X_train,y_train)
y_pred = clfnv.predict(X_test)
x = confusion_matrix(y_test,y_pred)
print(x)
sns.heatmap(x, annot=True)
print('Training Score {}'.format( clfnv.score(X_train,y_train)))
print('Testing Score {}'.format( clfnv.score(X_test,y_test)))
```

 # Training and Testing model with Logistic regression

```
clflg = LogisticRegression()
clflg.fit(X_train,y_train)
y_pred = clflg.predict(X_test)
x = confusion_matrix(y_test,y_pred)
print(x)
sns.heatmap(x, annot=True)
print('Training Score {}'.format(clflg.score(X_train,y_train)))
print('Testing Score {} '.format(clflg.score(X_test,y_test)))
```

EXPERIMENTAL RESULTS

Importing and Connection to DataBase:

Final Fundamentals of datascience Project

In the below cell I have imported all required libraries

In [1]:
```
from scipy import stats
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:
```
#Connecting to mysql database, AWS Server
import pymysql
conn=pymysql.connect(host='database-1.cjayyplhux0p.us-east-2.rds.amazonaws.com',port=int(3306),user='admin',passwd='Chinnu.969',
cursor=conn.cursor()
```

Displaying the Data set Information:

In [3]:
```
data=cursor.execute("select * from finaltable")
for row in cursor.fetchall():
    print (row)
```

```
(39, ' State-gov', ' Bachelors', 13, ' Never-married', ' Adm-clerical', ' Not-in-family', ' White', ' Male', 2174, 0, 40, ' U
nited-States', '<=50K')
(50, ' Self-emp-not-inc', ' Bachelors', 13, ' Married-civ-spouse', ' Exec-managerial', ' Husband', ' White', ' Male', 0, 0, 1
3, ' United-States', '<=50K')
(38, ' Private', ' HS-grad', 9, ' Divorced', ' Handlers-cleaners', ' Not-in-family', ' White', ' Male', 0, 0, 40, ' United-St
ates', '<=50K')
(53, ' Private', ' 11th', 7, ' Married-civ-spouse', ' Handlers-cleaners', ' Husband', ' Black', ' Male', 0, 0, 40, ' United-S
tates', '<=50K')
(28, ' Private', ' Bachelors', 13, ' Married-civ-spouse', ' Prof-specialty', ' Wife', ' Black', ' Female', 0, 0, 40, ' Cuba',
'<=50K')
(37, ' Private', ' Masters', 14, ' Married-civ-spouse', ' Exec-managerial', ' Wife', ' White', ' Female', 0, 0, 40, ' United-
States', '<=50K')
(49, ' Private', ' 9th', 5, ' Married-spouse-absent', ' Other-service', ' Not-in-family', ' Black', ' Female', 0, 0, 16, ' Ja
maica', '<=50K')
(52, ' Self-emp-not-inc', ' HS-grad', 9, ' Married-civ-spouse', ' Exec-managerial', ' Husband', ' White', ' Male', 0, 0, 45,
' United-States', '>50K')
(31, ' Private', ' Masters', 14, ' Never-married', ' Prof-specialty', ' Not-in-family', ' White', ' Female', 14084, 0, 50, '
United-States', '>50K')
(42, ' Private', ' Bachelors', 13, ' Married-civ-spouse', ' Exec-managerial', ' Husband', ' White', ' Male', 5178, 0, 40, ' U
```

In [22]:
```
from sqlalchemy import create_engine
import pymysql

db_connection_str = 'mysql+pymysql://admin:Chinnu.969@database-1.cjayyplhux0p.us-east-2.rds.amazonaws.com:3306/project'
db_connection = create_engine(db_connection_str)
```

In [23]:
```
df = pd.read_sql('SELECT * FROM finaltable', con=db_connection)
```

In [24]:
```
df
```

Out[24]:

	age	workclass	education_level	education_num	maritalstatus	occupation	relationship	race	sex	capitalgain	capitalloss	hoursperweek	nativec
0	39	State-gov	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40	United
1	50	Self-emp-not-inc	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13	United
2	38	Private	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40	United
3	53	Private	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0	0	40	United
4	28	Private	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0	40	
...	...	...	...	...	...	...	...	...	...	...	...	...	
45217	33	Private	Bachelors	13	Never-married	Prof-specialty	Own-child	White	Male	0	0	40	United
45218	39	Private	Bachelors	13	Divorced	Prof-specialty	Not-in-family	White	Female	0	0	36	United
45219	38	Private	Bachelors	13	Married-civ-spouse	Prof-specialty	Husband	White	Male	0	0	50	United
45220	44	Private	Bachelors	13	Divorced	Adm-clerical	Own-child	Asian-Pac-	Male	5455	0	40	United

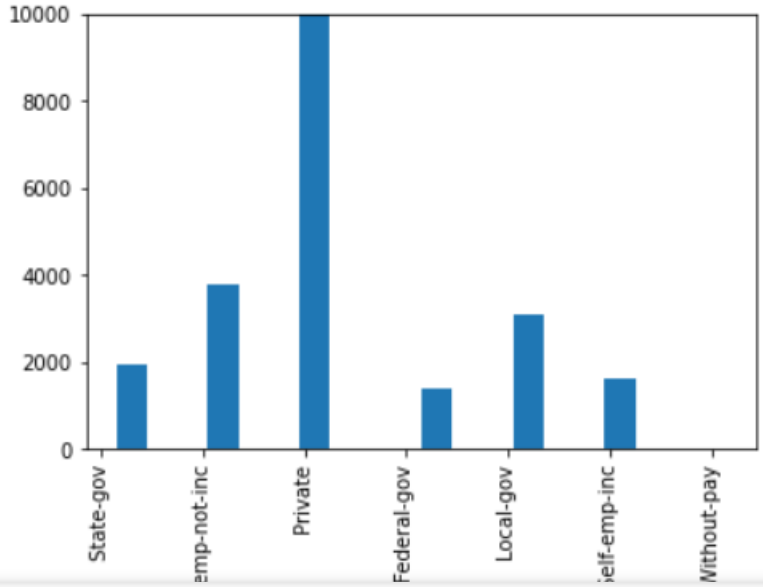
In [26]:
```
Features = df.columns
Features
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45222 entries, 0 to 45221
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   age              45222 non-null  int64
 1   workclass        45222 non-null  object
 2   education_level  45222 non-null  object
 3   education_num    45222 non-null  int64
 4   maritalstatus    45222 non-null  object
 5   occupation       45222 non-null  object
 6   relationship     45222 non-null  object
 7   race             45222 non-null  object
 8   sex              45222 non-null  object
 9   capitalgain      45222 non-null  int64
 10  capitalloss      45222 non-null  int64
 11  hoursperweek     45222 non-null  int64
 12  nativecountry    45222 non-null  object
 13  income           45222 non-null  object
dtypes: int64(5), object(9)
```

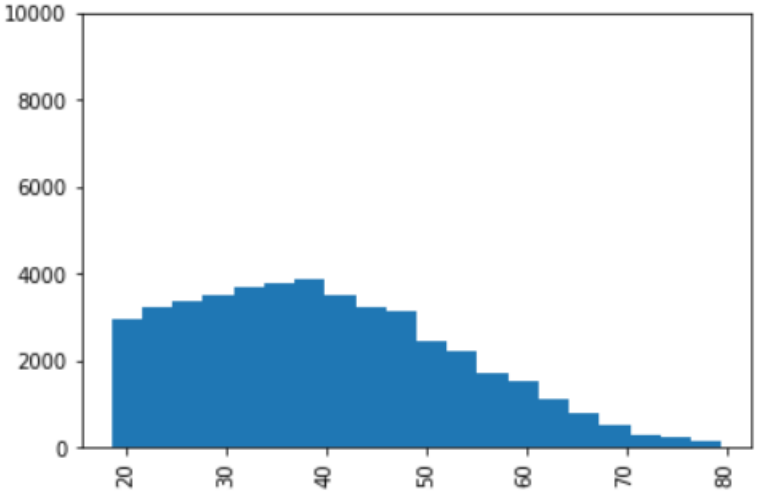
Data Quality Report for Continuous features

	feature	count	missing values(%)	Unique values	minimum value	1ST Quartile	mean	median	3RD Quartile	maximum value	standard deviation
0	age	45222	0.0	62	17	28.0	38.382469	37.0	47.0	78	12.930673
1	capitalloss	45222	0.0	14	0	0.0	5.107448	0.0	0.0	1258	35.894991
2	capitalgain	45222	0.0	116	0	0.0	552.340542	0.0	0.0	22040	2279.406498
3	hoursperweek	45222	0.0	71	5	40.0	40.503958	40.0	45.0	76	10.743984

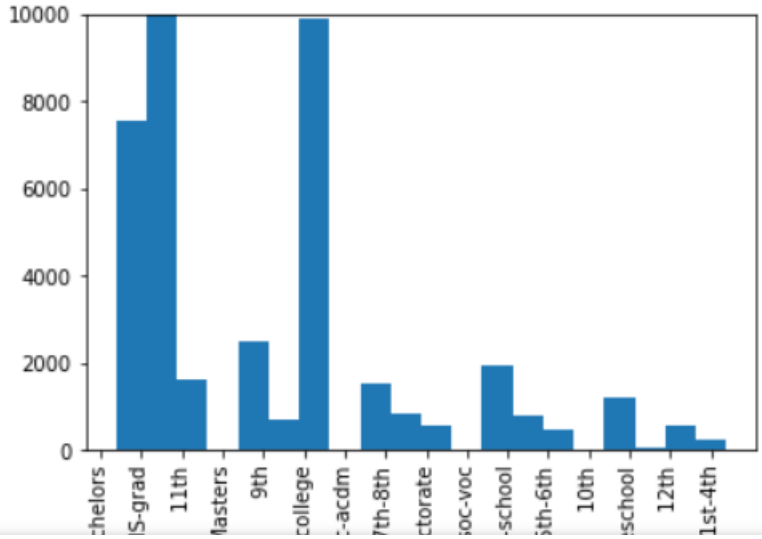
workclass



age



education_level



normalization of data:

In [13]:

```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
discriptive_features = data.drop('income',axis =1)
discriptive_DF = pd.DataFrame(data = discriptive_features)
numerical = ['age', 'education-num', 'capital-gain', 'capital-loss', 'hours-per-week']
normalized_DF = pd.DataFrame(data = discriptive_DF)
normalized_DF[numerical] = scaler.fit_transform(discriptive_DF[numerical])
display(normalized_DF.head(n = 5))
normalized_DF.to_csv('normalized_dataset.csv')
```

	age	workclass	education_level	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	native-country
0	0.360656	State-gov	Bachelors	0.800000	Never-married	Adm-clerical	Not-in-family	White	Male	0.098639	0.0	0.492958	United-States
1	0.540984	Self-emp-not-inc	Bachelors	0.800000	Married-civ-spouse	Exec-managerial	Husband	White	Male	0.000000	0.0	0.112676	United-States
2	0.344262	Private	HS-grad	0.533333	Divorced	Handlers-cleaners	Not-in-family	White	Male	0.000000	0.0	0.492958	United-States
3	0.590164	Private	11th	0.400000	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0.000000	0.0	0.492958	United-States
4	0.180328	Private	Bachelors	0.800000	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0.000000	0.0	0.492958	Cuba

In [2]:

```
# loading preprocessed data into dataframe using panda library
df = pd.read_csv('R:\Ram\Computer Science\FINAL PROJECT\Main\Dataset_After__pp.csv')
df = df.replace({'income': '>50K'},1)
df = df.replace({'income': '<=50K'},0)
data = df.copy()
data = data.drop('Unnamed: 0',axis = 1)
x = data.drop('income',axis=1)
y = data['income']
df
```

Out[2]:

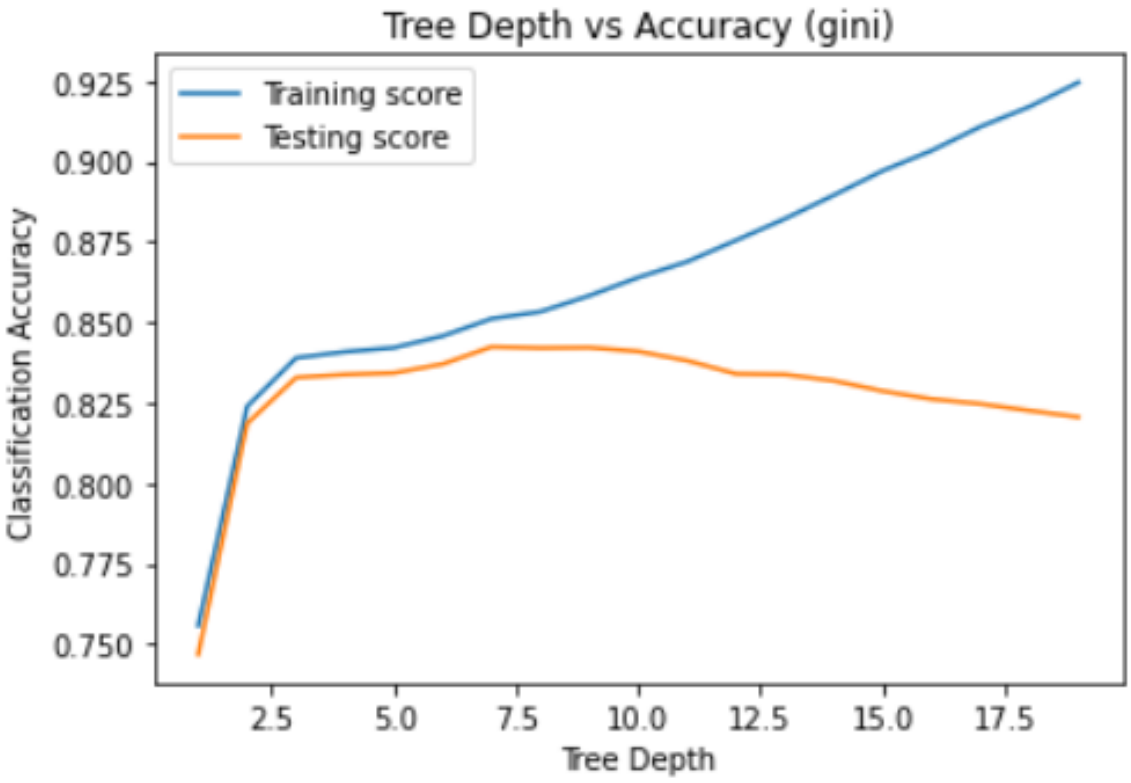
	Unnamed: 0	capital-gain	education-num	age	hours-per-week	relationship_ Husband	relationship_ Not-in-family	relationship_ Other-relative	relationship_ Own-child	relationship_ Unmarried	...	sex_ Female	sex_ Male	workclass_ Federal gov
0	0	0.098639	0.800000	0.360656	0.492958	0	1	0	0	0	...	0	1	0
1	1	0.000000	0.800000	0.540984	0.112676	1	0	0	0	0	...	0	1	0
2	2	0.000000	0.533333	0.344262	0.492958	0	1	0	0	0	...	0	1	0
3	3	0.000000	0.400000	0.590164	0.492958	1	0	0	0	0	...	0	1	0
4	4	0.000000	0.800000	0.180328	0.492958	0	0	0	0	0	...	1	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
45217	45217	0.000000	0.800000	0.262295	0.492958	0	0	0	1	0	...	0	1	0
45218	45218	0.000000	0.800000	0.360656	0.436620	0	1	0	0	0	...	1	0	0
45219	45219	0.000000	0.800000	0.344262	0.633803	1	0	0	0	0	...	0	1	0
45220	45220	0.247505	0.800000	0.442623	0.492958	0	0	0	1	0	...	0	1	0
45221	45221	0.000000	0.800000	0.295082	0.774648	1	0	0	0	0	...	0	1	0

45222 rows × 58 columns

Splitting the data set:

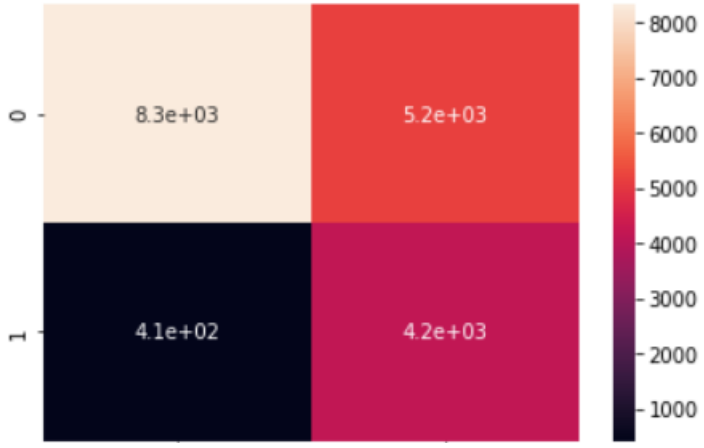
```
In [5]: # splitting data into testing and training data with 40% holdout
        X_train,X_test,y_train,y_test = train_test_split(x,y,test_size = 0.4)
```

DecisionTreeClassifie:



naivebayes:

```
[[8316 5193]
 [ 413 4167]]
Training Score 0.6858806619245936
Testing Score 0.6900878987229808
```

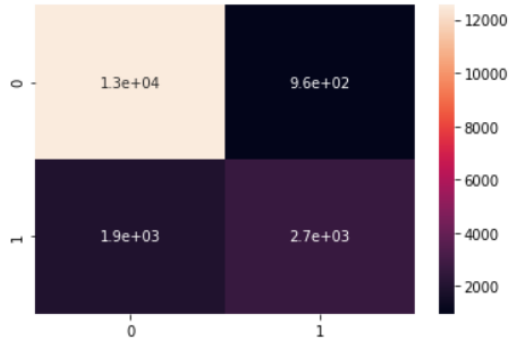


Logistic regression Output:

```
[[12544   965]
 [ 1925  2655]]
Training Score 0.8449858106364944
Testing Score 0.8402343965946155
```

```
c:\users\ramgo\appdata\local\programs\python\python38-32\lib\site-packages\sklearn\linear_model\_logistic.py:762: ConvergenceWa
rning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  n_iter_i = _check_optimize_result(
```



CONCLUSION AND FUTURE SCOPE

I got the charity's previous data sets. to classify users most likely to donate and estimate the donation amount. I cleaned the data, normalized and converted the necessary variables into numerical features to build predictive models. I shuffled and split our data into training and testing sets. I used 4 supervised machine learning methods, classification and regression models to identify future donors and estimate how much they will donate. These models were used to predict the outcome of the next campaign.

MODEL	TRANING ACCURACY(%)	TESTING ACCURACY(%)
DECISION TREE	84.7	84.5
KNN	85.8	82.8
NAVIE BAYES	73.4	73.3
LOGISTIC REGRESSION	84.4	84.1

The best suitable model for the prediction problem is the decision tree model with tree depth = 7, and the impurity measure is Entropy.

REFERENCES

[1] https://www.researchgate.net/publication/319852408_Concepts_and_Fundaments_of_Data_Warehousing_and_OLAP

[2] https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/two-class-logistic-regression

[3] MINING BIG DATA : DT4 -IEEE-IJSEAS

[4] Research paper classification systems-IEEE-Sang-Woon Kim & Joon-Min Gil

[5] DATA VISUALIZATION-Research Gate-Adebowale E. Shadare

[6] Concepts and Fundaments of Data Warehousing and OLAP -IEEE-Fernando Almeida

[7] A survey of data partitioning and sampling methods to support big data analysis-IEEE-Joshua Zhexue Huang

[8] An industrial big data pipeline for data-driven analytics maintenance applications in large-scale smart manufacturing facilities

[9] Design and Development of Data Pipelines-Karthik Cottur 1, Veena Gadad2

[10] Study and Analysis of Decision Tree Based Classification Algorithms

[11] Supervised Learning Algorithms of Machine Learning: Prediction of Brand Loyalty IEEE